

Amendments to the Claims:

Please amend the claims as follows:

1. (Currently Amended) A process comprising:

providing a network filesystem on a client;

wherein said network filesystem handles and forwards requests from steaming-enabled local processes on said client that are directed at streaming software files located on said a server;

wherein said network filesystem examines said requests, and either grants or denies each of said requests depending on whether the request is justifiable from a security perspective by using information that includes, but is not limited to: the nature of the originating streaming-enabled process, the history of previous access by the streaming-enabled process, and/or the section of the targeted streaming software file being requested;

providing a network redirector component of said network filesystem; and

wherein said network redirector component makes visible to said network filesystem, a path that represents the server where said streaming software files are stored.

2. (Previously Presented) The process of claim 1, wherein said network filesystem registers dispatch routines with the client operating system that handle zero or more common file operations selected from the group consisting of open, read, write and close; wherein a dispatch routine examines a file request and decides whether to grant or deny said file request; and wherein if said file request is granted then said dispatch routine forwards said file request to said server and sends back said server's response to said client operating system.

3. (Previously Presented) The process of claim 1, wherein when a local steaming-enabled process on said client makes a file request for a streaming software file on said server, said client operating system calls a dispatch routine with said file request.

4. (Cancelled)

5. (Cancelled)

6. (Cancelled)

7. (Cancelled)

8. (Cancelled)

9. (Cancelled)

10. (Previously Presented) A program storage medium readable by a computer, tangibly embodying a program of instructions executable by the computer to perform method steps for preventing the piracy of application programs resident on a server and remotely accessed across a computer network by a client system in a computer environment, comprising the steps of:

providing a network filesystem on said client;

wherein said network filesystem handles and forwards all requests from streaming-enabled local processes on said client that are directed at streaming software files located on said server;

wherein said network filesystem examines each of said requests, and either grants or denies each of said requests depending on whether the request is justifiable from a security perspective by using information that includes, but is not limited to: the nature of the originating streaming-enabled process, the history of previous access by the streaming-enabled process, and/or the section of the targeted file streaming application program being requested;

providing a network redirector component of said network filesystem; and

wherein said network redirector component makes visible to said network filesystem, a path that represents the server where said streaming software files are stored.

11. (Previously Presented) The method of claim 10, wherein said network filesystem registers dispatch routines with the client operating system that handle zero or more common file operations selected from the group consisting of open, read, write and close; wherein a dispatch routine examines a file request and decides whether to grant or deny said file request; and wherein if said file request is granted then said dispatch routine forwards said file request to said server and sends back said server's response to said client operating system.

12. (Previously Presented) The method of claim 10, wherein when a local steaming-enabled process on said client makes a file request for a streaming software file on said server, said client operating system calls a dispatch routine with said file request.

13. (Cancelled)

14. (Cancelled)

15. (Cancelled)

16. (Cancelled)

17. (Cancelled)

18. (Cancelled)

19. (Previously Presented) A process comprising:

providing a filesystem on said client; wherein said filesystem handles and forwards file requests from streaming enabled local processes on said client;

wherein said filesystem examines said requests, and either grants or denies each of said requests depending on whether the request is justifiable from a security perspective by using information that includes, but is not limited to: the nature of the originating streaming enabled process, the history of previous access by the streaming enabled process, and/or the section of the targeted streaming software file being requested;

wherein said filesystem registers dispatch routines with the client operating system that handle zero or more common file operations selected from the group consisting of open, read, write and close;

wherein a dispatch routine examines a file request and decides whether to grant or deny said file request; and

wherein if said file request is granted, then said dispatch routine allows the requested operation to proceed.

20. (Cancelled)

21. (Cancelled)

22. (Cancelled)

23. (Cancelled)

24. (Cancelled)

25. (Previously Presented) A program storage medium readable by a computer, tangibly embodying a program of instructions executable by the computer to perform method steps for preventing the piracy of application programs resident on a client system in a computer environment, comprising the steps of:

providing a filesystem on said client;

wherein said filesystem handles and forwards all file requests from streaming enabled local processes on said client;

wherein said filesystem examines each of said requests, and either grants or denies each of said requests depending on whether the request is justifiable from a security perspective by using information that includes, but is not limited to: the nature of the originating streaming enabled process, the history of previous access by the streaming enabled process, and/or the section of the targeted streaming software file being requested;

wherein said filesystem registers dispatch routines with the client operating system that handle zero or more common file operations selected from the group consisting of open, read, write and close;

wherein a dispatch routine examines a file request and decides whether to grant or deny said file request; and

wherein if said file request is granted, then said dispatch routine allows the requested operation to proceed.

26. (Cancelled)

27. (Cancelled)

28. (Cancelled)

29. (Cancelled)

30. (Cancelled)

31. (Cancelled)

32. (Cancelled)

33. (Cancelled)

34. (Cancelled)

35. (Previously Presented) A system comprising:

a processing device for processing a request for access to streaming software files stored on at least one server system that is remote from said processing device;

a redirector component that is associated with said processing device for informing said processing device of one or more locations in which said streaming software files are stored; and

wherein said processing device comprises a component that determines whether to grant requests for access to said streaming software files based on: whether an originating process that is making said requests for access is a trusted process, whether a history of previous requests for access made by said originating process exhibits a pre-determined pattern of piracy, and whether a section of said streaming software files that is being requested is a critical section that requires protection from piracy.

36. (Previously Presented) A system comprising:

a processing means for processing requests for access to streaming software files stored remotely from said processing means;

a redirection means for revealing one or more locations in which said streaming software files are stored; and

wherein said processing means includes a determination means for determining whether to grant requests for access to said streaming software files based on: whether an originating process that is making said requests for access is a trusted process, whether a history of previous requests for access made by said originating process exhibits a pre-determined pattern of piracy, and whether a section of said streaming software files that is being requested is a critical section that requires protection from piracy.

37. (Previously Presented) A system comprising:

a filtering means for filtering requests for access to streaming software application program files stored remotely from said filtering means;

a redirection means for revealing one or more locations in which said requested streaming software application program files are stored; and

wherein said filtering means includes an evaluation means for evaluating: an originating process that is making said requests for access, a history of previous requests for access made by said originating process, and a section of said streaming software application program files that is being requested.

38. (Cancelled)

39. (Cancelled)

40. (Previously Presented) A computer-readable medium carrying one or more sequences of instructions for preventing piracy of application program files in a computer system, wherein execution of the one or more sequences of instructions by one or more processors causes the one or more processors to perform:

providing information relating to one or more remote locations where streaming software files are stored; and

determining whether an originating process that is making said requests for access is a trusted process, whether a history of previous requests for access made by said originating process exhibits a pre-determined pattern of piracy, and whether a section of said streaming software files that is being requested is a critical section that requires protection from piracy.

41. (Previously Presented) A system comprising:

a means for providing location information to a local computing system of streaming software files that are stored on one or more remote locations;

a means for examining requests for access to said streaming software files;

a means for determining whether said requests can be granted based on whether an originating process that is making said requests for access is a trusted process, whether a history of previous requests for access made by said originating process exhibits a pre-determined pattern of piracy, and whether a section of said streaming software files that is being requested is a critical section that requires protection from piracy; and

a means for forwarding said requests to a corresponding server that is responsible for serving said streaming software files if said requests are granted.

42. (Previously Presented) A method comprising the computer-implemented acts of:

providing information relating to one or more remote locations where streaming software application program files are stored;

receiving a request from a computer process for access to said streaming software application program files;

determining if said computer process that is making said request for access is a trusted process; and

if said computer process is a trusted process, then forwarding said request to a corresponding remote server that is responsible for serving said streaming software application program files.

43. (Previously Presented) A method comprising the computer-implemented acts of:

providing information relating to one or more remote locations where streaming software files are stored;

receiving a request from a computer process for access to said streaming software files;

determining if a history of previous requests for access made by said computer process lacks a pre-determined pattern of piracy; and

if history of previous requests of said computer process lacks a pre-determined pattern of piracy, then forwarding said request to a corresponding remote server that is responsible for serving said streaming software files.

44. (Previously Presented) A method comprising the computer-implemented acts of:

providing information relating to one or more remote locations where streaming software application program files are stored;

receiving a request from a computer process for access to a section of said streaming software application program files;

determining if said section that is being requested is a non-critical section; and

if said section is a non-critical section, then forwarding said request to a corresponding remote server that is responsible for serving said streaming software application program files.

45. (New) The method of claim 1, further comprising using a filtering mechanism that is associated with the client for filtering requests for access to the streaming software files.

46. (New) The method of claim 1, further comprising providing information relating to one or more remote locations where streaming software files are stored.

47. (New) The method of claim 1, further comprising:

providing information relating to one or more remote locations, including the server, where streaming software files are stored;

using dispatch routines for examining a request for access to said streaming software files;

after examining said request and if it is determined that a history of previous requests for access made by said originating process lacks a pre-determined pattern of piracy or that a section of said streaming software files that is being requested is a non-critical section, then forwarding said request to the server.

48. (New) The method of claim 1, further comprising:

using a filtering mechanism on the client for filtering requests for access to streaming software files;

using a revealing mechanism to reveal to said client one or more remote locations, including the server, on which said requested streaming software files are stored.